Serial Number:

University of Bahrain
College of Information Technology
Department of Computer Science
Semester I, 2014-2015
ITCS102/ITCS104/ ITCS112 (Computer Programming II)

TEST 2

Date: 22/12/2014 Time: 11:00 - 12:15

STUDENT NAME	ANSWER
STUDENT ID #	
SECTION #	

QUESTION #	MARKS		COMMENTS
Q.1	6		
Q.2	10		
Q.3	7		
Q.4	10		
Q.5	7		
TOTAL	**40**		

Question 1 [6 points]

Show the output of the following C++ program:

```
#include<iostream>
using namespace std;


int main ( )
 {
  double *p,*q;
   double x=3.0, y =1.5;
   p=&x;
  q=new double;
  *q= *p;
  cout<<*p<<" "<<*q<<endl;
   x+=y;
   cout<<*p<<" "<<*q<<endl;
   p=q;
   *p= *q + y;
    cout<<*p<<" "<<*q<<endl;
    return 0;
}
```

OUTPUT

Each 1 pt = Total = 6pts

3 3
4.5 3
4.5 4.5

Question 2 [10 points]

Create a class named **Triangle,** with three sides ***a***, ***b***, and ***c*** as its data members. Each of the three sides of the triangle is a positive real number with a default value of 1.0.

The class should include the following member functions:

1. A set function to set all data members.
2. Three get functions named *getSideA*, *getSideB*, *getSideC* to return the values of a, b and c respectively
3. A function to print the values of the data members.
4. A constructor with default value paramaters.
5. A destructor function to output a message "A Triangle object is deleted".

Note: Declare the class only, don't include the mmeber functions implementation.

```
// Total = 10 pts
class Triangle{ // 1 pt
    private:  // 1 pt
       float a;  // 1.5 pts
       float b;
       float c;

    public:   // 1 pt
       Triangle(float=1.0, float=1.0, float=1.0); // 1 pt
      ~Triangle(); // 1 pt
       void  setTriangle(float, float, float); // 1 pt
       void getSideA(float&); // 0.5 pt
       void getSideB(float&); // 0.5 pt
       void getSideC(float&);  // 0.5 pt
       void print(); // 1 pt
};
```

Question 3 [7 points]

Write the definition (Implementation) of the ***set*** function defined in Question.2

```
//Total = 7 pts
void Triangle :: setTriangle(float s1, float s2, float s3) // header 1 pts
{
// each 2 pts; 1 pt for set, and 1 pt for validation  -> Total = 6 pts

  if(s1>0) a=s1; else a =1.0;

  if(s2>0) b=s2; else b =1.0;

  if(s3>0) c=s3; else c =1.0;

}
```

Question 4 [10 points]

Write a function named **countEquilateralTriangles** that takes as paramaters: a pointer to an array of type **Triangle**, and the array size. The function should count and return the number of **equilateral triangles** in the array**.** In geometry, an **equilateral triangle** is a triangle in which all three sides are equal. The function prototype is:

*int countEquilateralTriangles (Triangle * list, int length);*

```
int countEquilateralTriangles (Triangle * list, int length){
 float a, b, c;
 int count=0;  // declare count, count ++, return 2 pts

 for(int i=0; i<length; i++) // 2 pts
   {
    list[i].getSideA(a); // 3 pts for using get functions + declare a,b,c
    list[i].getSideB(b);
    list[i].getSideC(c);

    if(a==b && b==c)  // 3 pts
      count++;
   }
   return count;
}
```

Question 5 [7 points]

Write a main function to test the function **countEquilateralTriangles** defined in Question-4 using one-dimensional ***dynamic array***. The size and the contents of the array should be entered by the user.

```
int main(){
    //Total = 7 pts

   // read len = 1 pt
   int len;
   cout<<"Enter # of elements:";
   cin>>len;

   // create a dynamic array 2 pt
   Triangle * list;
   list = new Triangle [len];


   //read sides of the triangles = 2 pts
    float a,b,c;
   for(int i=0; i< len; i++)
     {
      cout<<"Enter a, b, and c sides: ";
      cin>>a>>b>>c;

      list[i].setTriangle(a,b,c);

     }

   // call and output # of Equilateral Triangles 2 pts
   int count= countEquilateralTriangles(list, len);

   cout<<"The number of Equilateral Triangles = "<<count<<endl;


   system("pause");
   return 0;
}
```